LA-UR -84-3942

Conf-850255--1

Los Alamos National Laboratory is operated by the University of California for the United States Department of Energy under contract W-7405-ENG-36

LA-UR--84-3942

DE85 005574

TITLE: BOTTLENECKOLOGY: EVALUATING SUPERCOMPUTERS

AUTHOR(S): Jack Worlton

MASTER

SUBMITTED TO: IEEE Spring COMP/CON, February 1985, San Francisco, CA

## DISCLAIMER





Los Alamos Los Alamos National Laboratory
Los Alamos, New Mexico 87545

FORM NO 836 R4
ST NO 2629 5/81

# BOTTLENECKOLOGY: EVALUATING SUPERCOMPUTERS

Jack Worlton

Computing and Communications Division
Los Alamos National Laboratory
Los Alamos, New Mexico 87545

## ABSTRACT

Evaluating supercomputer performance is more difficult than evaluating performance for other types of computers because of the wide range of performances encountered. Depending on the purpose of the evaluation, methods of evaluation can be used that trade off level of effort and accuracy, including rules of thumb, analytical models, testing, and simulation.

## 1. INTRODUCTION

There are two motives for the evaluation of supercomputers: to predict the performance of alternative architectures during the design process and to predict the performance of alternative computers during the acquisition process. The level of effort expended on the first of these motives is typically much greater than for the second. This paper focuses on the methods used in the acquisition process, although the methods are generally applicable.

The general methods used for system evaluation include rules of thumb, analytical models, simulation, and testing.[1] Rules of thumb are the easiest to apply but give the least accurate results; simulation and testing are the most expensive but give the most accurate results. It is usually best to apply the methods in top-down order, beginning with rules of thumb to get ballpark performance estimates, analytical models to gain deeper insights, and then continuing with more detailed studies using simulation and testing. Simulation studies are usually considered to be too demanding for the acquisition process, and they will not be covered in this paper.

## 2. SOME RULES OF THUMB

Rules of thumb are judgments formed from experience and hence vary widely with the people who apply them. The goal of a rule of thumb is to gain a good approximation to supercomputer performance using the minimum information, time, and effort. Rules of thumb are sufficiently accurate to determine peak performance rates and to give an estimate of expected performance. For this purpose, the key item of information is the machine cycle time, Tc. The reciprocal of Tc is usually the maximum rate at which the machine will issue instructions; hence, one rule of thumb is that 1/Tc is the peak "no-op" rate, i.e., the maximum rate of execution of scalar instructions.

The sustained rate of scalar execution is, of course, much less than this, due to delays in memory access and operation execution. For example, the CDC 7600 had a cycle time of 0.0275 microseconds, hence its no-op rate was about 36 million instructions per second (MIPS); however its sustained rate in Fortran codes was only about 10 MIPS, or 28% of its no-op rate. Architectural efficiency in different designs will cause this number to vary somewhat, but as a rule of thumb we expect most computers to execute scalar code at about one-third of their no-op rate, i.e., 1/(3*Tc).

It is sometimes of interest to know the rate of executing floating-point operations. For this purpose a rule of thumb is used for converting MIPS to millions of floating-point operations per second (MFLOPS). Estimates vary from 3 to 5 for the ratio of MIPS/MFLOPS, but 4 is commonly used. Thus, given Tc, we can estimate the scalar MFLOPS rate by 1/(12*Tc). To take the CDC 7600 again, this would give 1/(12*0.0275) = 3 MFLOPS, which is in the right ballpark.

If a range of scalar performance is known, the weighted harmonic mean can be used to estimate sustained performance. Suppose, for example, we know that the CDC 7600 has a performance range of 2 to 6 MFLOPS and that these are to be equally weighted; the harmonic mean is then given by

$$Hm = 2/(1/2 + 1/6) = 3 \text{ MFLOPS}.$$

Rules of thumb for vector performance are less precise than those for scalar performance because of the wide range of vector performance. It is not unusual for the peak to low vector performance range to be as high as 50:1,[2] whereas scalar performance ranges are typically less than 5:1. Peak vector rates can be computed from the ratio of the number of results generated per cycle divided by the time per cycle. For example, the Cyber 205 with four pipelines and a cycle time of 0.020 microseconds has a peak vector performance of 4/0.020 = 200 MFLOPS; this computer also has a "triadic accelerator" that can double this performance, so peak performance would then go to 400 MFLOPS.

We can now combine our scalar and vector rules of thumb for a composite estimate by using the harmonic mean. Suppose a vector processor has a sustained rate of 50 MFLOPS and a sustained scalar rate of 5 MFLOPS. The harmonic mean of these rates (using equal weights) would then be

$$Hm = 2/(1/50 + 1/5) = 9 \text{ MFLOPS}.$$

In summary, to obtain a quick estimate of the performance of a supercomputer, estimate sustained scalar performance from the cycle time, sustained vector performance from the peak vector rate, and then combine these using the harmonic mean.

## 3. ANALYTICAL MODELS

Analytical models are more difficult to derive than rules of thumb, but once they are understood, they can be used with relatively little effort; models can be readily coded up and run in Basic on a personal computer, for example. The theoretical foundation of these models is the weighted harmonic mean:

$$Hm = \frac{1}{\sum_i f_i/R_i},$$

where $f_i$ = the fraction of results generated at rate $R_i$. This model is so basic to understanding computer performance that it might be called *The Fundamental Principle of Computer Architecture*. It is also the basis of what is called "bottleneckology"--the study of bottlenecks. This generic model can be used to derive architecture-specific models.

The analytical model commonly called "Amdahl's Law" is a special case of the weighted harmonic mean that includes only the lowest and highest performances being considered, i.e.,

$$A = \frac{1}{f/R + (1-f)/r},$$

where f is the fraction of work done at the high rate, R, and (1-f) is the fraction of work done at the low rate, r.[3]

A third analytical model useful for performance estimation is the uniform average.* If we assume that the set of fractions of vectorization for a site workload are uniformly distributed over the interval (0,1), we can apply the uniform average from the integral calculus to Amdahl's Law, and derive the following result:

$$Ra = Rv*(\ln Rvs)/(Rvs-1),$$

where Ra = the uniform average rate, Rv = the peak rate, and Rvs = the ratio of peak to minimum rate. The factor (ln Rvs)/(Rvs-1) is essentially an efficiency factor that determines what fraction of the peak performance, Rv, will be achieved for a given ratio of peak to low performance.

This result can be generalized by taking the uniform average over the interval (a,b) within the interval (0,1).

## 4. TESTING

Testing of supercomputers, commonly called "benchmarking," poses a challenge because of the wide performance range of supercomputers. A small variation in the optimization of a code can cause a significant difference in the benchmark results. Thus, benchmarks are site-dependent: benchmarks for Site A may not represent the performance that would be achieved by Site B. To assure reliable benchmark data, a site should follow at least the following steps:

- o Perform a workload characterization study. This will define the types of jobs and the fraction of the workload they represent.
- o Select a subset of the jobs to represent the whole workload; these should be those jobs having the highest fractions in the workload.
- o Select portions of these programs (kernels) to represent the whole program. This is perhaps the most crucial step in the process.
- o Time the kernels on the systems of interest to obtain kernel execution rates.
- o Compute the weighted harmonic mean using the fractions obtained in the workload characterization study.

Failures in the benchmarking process are usually due to the lack of a workload characterization study, to the use of kernels that do not represent the programs, or to the use of the arithmetic mean rather than the harmonic mean.

## 5. SUMMARY

New generations of supercomputers pose ever more difficult evaluation problems due to the innovative architectures in these computers. The new class of supercomputers that will use multiple instruction-stream designs will require even more careful evaluation procedures than have been used in the past.

## 6. REFERENCES


1. Kishor Trivedi, "Analysis of Computer Performance and Reliability," *Proc. Int. Conf. on Computer Design*, 1983, pp. 464-467.
2. Jack Worlton, "Understanding Supercomputer Benchmarks," *Datamation*, September 1, 1984, pp. 121-130.
3. Jack Worlton, *A Philosophy of Supercomputing*, Los Alamos National Laboratory report LA-8849-MS, June 1981.


*I am indebted to Dr. Leland Williams of Triangle Universities Computing Center for this insight; he, in turn, credits it to Professor R. Cusson of Duke University.